Title of the Invention

A failure information management method and management server in a network equipped with a storage device

Background of the Invention

The present invention relates to a computer system equipped with a storage system. More specifically, the present invention relates to a method and device for managing storage system failures in storage area networks (Storage Area Network, hereinafter referred to as SAN) in which real volumes from a storage system are provided by a server as virtual volumes.

(1) SANs

In recent years, SANs are becoming more widespread. A SAN is a network dedicated to storage I/O in which storage is consolidated and separated from servers. The deployment of SANs makes it possible to provide high-speed data transfer, high scalability and availability in storage system, and efficient usage of storage resources.

(2) SAN management

The high scalability of SAN-based storage systems allows devices (servers, switches, storage devices) from multiple vendors to be mixed in a SAN. Using SAN without disruption requires SAN management.

An overview of SAN management is described in p. 331 - 334 of "Storage Area Network Essentials", John Wiley & Sons, Inc by Richard Barker and Paul Massiglia. In SAN management, availability monitoring of the devices connected in the SAN is especially important and is the basis of day-to-day operation. The software used for monitoring SAN availability is hereinafter referred to as the SAN manager.

A SAN manager has two major functions: configuration management and failure monitoring.

In the configuration management function, information is periodically obtained from a management agent present in each device connected in the SAN, the physical connections (topology) of the SAN is determined from the obtained information, and an up-to-date topology is continuously visualized and provided to the user of the SAN manager, i.e., the SAN administrator.

In failure monitoring, event notifications issued from devices connected in the SAN, e.g., notifications of hardware failure or performance decreases, and device information periodically obtained from the management agent present in each device are used to detect events such as failures and performance drops, and these events are notified to the SAN administrator.

With these two functions, the SAN administrator is able to use the SAN manager to provide unified management over availability, thus allowing reductions in operating costs, e.g., through reducing the number of SAN administrators.

(3) Virtualization devices

Virtual volumes technology is a SAN storage management technology. Virtual volume technology is disclosed in U.K. laid-open patent application number 2351375. The UK laid-open patent application number 2351375 discloses a device referred to as a storage server that has the following two functions.

1) A function for managing a volume (hereinafter referred to as real volume), which is a storage region in a storage medium in a storage device connected to the storage server, and for generating a volume pool.

2) A function for generating a virtual volume from at least one real volume in a volume pool, sequentially converting virtual volume I/O requests from the server to real volume I/O requests, and responding to I/O requests from the server.

The device having these two functions will be referred to hereinafter as a storage virtualization device. By using a virtualization device in a SAN, volume allocations to the server can be centralized using virtual volumes, thus eliminating the need to be aware of the physical arrangement of the storage devices connected to the virtualization device. In other words, the SAN administrator can allocate volumes in a centralized manner.

Summary of the Invention

With the failure monitoring function of the SAN manager, the SAN administrator identifies the device and component that is causing a failure based on events issued from multiple devices. This will be referred to hereinafter as "failure investigation". By having the virtualization device provide virtual volumes, a greater degree of freedom is possible for the configuration of volumes provided to the server. However, investigating a failure based on failure messages (SNMP Traps and the like) issued by devices from multiple vendors connected to the SAN depends on the SAN administrator having a high degree of knowledge about individual devices. This increases management costs.

Also, the SAN manager has a failure notification function which performs event notification to management software managing the overall enterprise system (hereinafter referred to as the high-level system management software), sending e-mail to the SAN administrator, and the like, depending on the severity of the failure. However, since the definition of the severity of a failure is dependent on the individual devices connected to the SAN, for each failure the SAN administrator must judge whether a particular event from a particular device has a high severity. Thus, responding to failure becomes time consuming.

A first object of the present invention is to support failure investigation by the SAN administrator when a failure message is issued from a device connected to the SAN.

A second object of the present invention is to allow the SAN administrator and high-level system management software to receive, out of the failure messages issued by the devices connected to the SAN, the failure information that is necessary.

To achieve the first object, a management server receiving multiple failure notifications from devices connected to a SAN associates and outputs multiple failure notifications based on association relationships between real volumes and virtual volumes managed by the virtualization device.

To achieve the second object, a management server receiving multiple failure notifications from devices connected to a SAN takes information indicating the severity of failure information, based on different standards, contained in the failure notifications and converts this information to severity information based on a common standard, and then processes the failure notification based on the converted severity.

Brief Description of the Drawings

Fig. 1 is a drawing showing a sample SAN architecture.

Fig. 2 is a drawing showing a sample architecture of a SAN management server.

Fig. 3 is a drawing showing a sample architecture of a server.

Fig. 4 is a drawing showing a sample architecture of a switch.

Fig. 5 is a drawing showing a sample architecture of a storage device having a storage virtualization function.

Fig. 6 is a drawing showing a sample architecture of a storage device.

Fig. 7 is a drawing showing a sample real volume mapping management table stored in a SAN management server.

Fig. 8 is a drawing showing a sample virtual volume mapping management table stored in a SAN management server.

Fig. 9 is a drawing showing a sample device detection list stored in a SAN management server.

Fig. 10 is a drawing showing a sample data interface management table stored in a server.

Fig. 11 is a drawing showing a sample volume management table stored in a server.

Fig. 12 is a drawing showing a sample FC connection management table stored in a switch.

Fig. 13 is a drawing showing a sample real volume management table stored in a storage device.

Fig. 15 is a drawing showing a sample virtual volume management table stored in a storage device.

Fig. 16 is a drawing showing a sample failure analysis dictionary table associated with a storage device and stored in a SAN management server.

Fig. 17 is a drawing showing a sample failure log stored in a SAN management server.

Fig. 18 is a drawing showing a sample failure severity conversion table stored in a SAN management server.

Fig. 19 is a flowchart showing an example of operations performed by a SAN management server to generate real topology mapping and virtual topology mapping for a storage network.

Fig. 20 is a detailed flowchart showing an example of operations performed by a SAN management server to generate a virtual volume mapping management table.

Fig. 21 is a detailed flowchart showing an example of operations performed by a SAN management server to generate a real volume mapping management table.

Fig. 22 is a flowchart showing an example of operations performed by a SAN management server to investigate failures.

Fig. 23 shows a sample failure investigation results display output by a SAN management server.

Fig. 24 is a flowchart showing an example of failure notification operations including a severity conversion function performed by a SAN management server.

Fig. 25 is a drawing showing a sample structure of an SNMP Trap message.

Fig. 26 is a drawing showing a sample configuration of a SAN.

Fig. 27 is a drawing showing a sample configuration of a SAN.

Description of the Preferred Embodiment

The embodiments of the present invention will be described below, with references to the figure. The present invention is not restricted to these embodiments.

<SAN architecture>

First, a sample SAN architecture according to this embodiment will be described. Fig. 1 through Fig. 6 show sample architectures of a SAN and devices connected to the SAN. Fig. 9 through Fig. 18 show management information contained in the devices.

Fig. 1 shows a sample SAN architecture. The SAN of the present invention includes: at least one server having a management agent; at least one switch having a management agent; at least one virtualization device having a management agent; at least one storage device having a management agent; and a single SAN management server having a SAN manager.

For convenience in the following description, the SAN of this embodiment includes: a single server (server A) 20000, a single switch (switch A) 30000, a single storage device with virtual volume technology (storage device A) 40000, a single storage device (storage device B 50000). These elements are connected to each other by a Fibre Channel 60000. In this embodiment, the storage device A40000 recognizes a real volume 57000 of the storage device B50000 by way of the switch 30000. Using the virtual volume function of the storage device A40000, the real volume 57000 of the storage device B50000 is provided to the server as a virtual volume of the storage device A40000.

Regarding how the storage device A40000 and the storage device B50000 are connected, this connection does not have to go through the switch A30000 as shown in the example in Fig. 1. For example, as in the second sample architecture shown in Fig. 26, the Fibre Channel 60000 can be used to directly connect the storage device A40000 and the storage device B50000. Also, as shown in the third sample architecture in Fig. 27, there can be a combination of a path that directly connects the storage device A40000 with the storage device B50000 and a path in which the storage devices are connected by way of a switch.

A SAN management server 10000 is connected by way of a management network 70000 to the server, the switch, and the storage devices. The management agent of each device and a SAN manager 13100 in the SAN management server 10000 can communicate by way of the management network. The SAN manager 13100 performs operations described later to manage the configuration of virtual volumes and real volumes, to investigate failures in the SAN, and to perform failure notifications in the SAN.

The memory 13000 stores: a SAN manager 13100, which is a program executed by the SAN management server; a real volume mapping management table 13200 storing real volume mapping information for the SAN; a virtual volume mapping management table 13300 storing virtual volume mapping information for the SAN; a real topology repository 13400, which is a memory area storing information collected from management agents in the devices in the SAN; a device detection list 13500 storing a list of devices in the SAN to be managed by the SAN manager 13100; at least one failure analysis dictionary 13600 for analyzing failure notification message contents received from the devices in the SAN; a failure log 13700 for recording event contents; and at least one failure severity conversion table 13800, used to perform severity

conversion, described later, storing severity conversion definitions defined ahead of time by a SAN administrator.

Fig. 3 shows a sample architecture of the server 20000. The server 20000 includes: a processor 21000; a memory 23000; a management interface 24000 for connecting to the management network 70000; and at least one data interface 26000 for connecting to the Fibre Channel 60000. These elements are connected to each other by a communication path 27000, e.g., an internal bus.

The memory 23000 stores: a management agent 23100, which is a program for communicating with the SAN manager 13100 to send and receive the server's management information; a data interface management table 23200 storing management information for the server's data interface; and a volume management table 23300 storing management information for volume's that the server accesses.

In this embodiment, there is one server, server A, and the server A is equipped with one data interface. However, the number of servers and data interfaces does not have to be one. Multiple servers can be connected to the SAN and a single server can be equipped with multiple data interface. Each data interface in a server is assigned an identifier (data interface ID) that is unique within the server. In this embodiment, the data interface ID of the server A is a1.

Fig. 4 shows a sample architecture of the switch 30000. The switch 30000 includes: a controller 31000 performing switching of data sent to and received by way of the Fibre Channel 60000; a memory 33000; a management interface 34000 for connecting to the management network 70000; and at least one data interface 36000 for connecting to the Fibre Channel 60000. The memory 33000, the management interface 34000, and the data interface 36000 are connected to each other by the controller 31000.

The memory 33000 stores: a management agent 33100, which is a program for communicating with the SAN manager 13100 to send and receive management information for the switch A; and an FC connection management table 33200, which contains information indicating how the switches, the server, and the storage devices are connected by the Fibre Channel.

In this embodiment, there is a single switch A in the SAN, and this switch A includes six data interfaces. However, any number of switches and data interfaces can be used. Each data interface has an identifier (data interface ID) that is unique within the switch. In this embodiment, these values are s1, s2, s3, s4, s5, and s6.

Fig. 5 shows a detailed drawing of a sample architecture of the storage device A, which is a storage device with the virtual volume technology. The storage device A 40000 includes: a controller 41000 providing internal control for the storage device; a memory 43000; a management interface 44000 for connecting to the management network 70000; at least one data interface 46000 for connecting to the Fibre Channel 60000; and at least one real volume 47000, which stores data used by the server and has a storage region inside the storage device A. The memory 43000, the management interface 44000, the data interface 46000, and the real volume 47000 are connected to each other by the controller 41000.

In addition to the real volume 47000 serving as a storage region connected to the controller 41000, Fig. 5 also shows at least one virtual volume 48000. This virtual volume 48000 is a virtual entity created by the volume virtualization function of the storage device A from a real volume stored in another storage device (e.g., the storage device B50000). The virtual volume 48000 is indicated in Fig. 5 as an element of the

storage device A since it is a storage region provided to the server as a volume of the storage device A, but the actual storage region exists in another storage device connected to the storage device A.

The memory 43000 stores: a management agent 43100, which is a program for communicating with the SAN manager 13100 to send and receive management information of the storage device A; a data interface management table 43200 stores management information for the data interface of the storage device A; a real volume management table 43300 stores management information for the real volume 47000 of the storage device A; a virtual volume management program 43400 implementing the volume virtualization function; and a virtual volume management table 43500 storing virtual volume management information provided to the servers by the storage device.

In this embodiment, the storage device A includes two data interfaces and two real volumes and two virtual volumes. However, there can be any number of data interfaces, real volumes, and virtual volumes. Each data interface, real volume, and virtual volume has an identifier (data interface ID, real volume ID, virtual volume ID) that is unique within the device. In this embodiment, the data interface IDs are c1, c2, the real volume IDs are va1, va2, and the virtual volume IDs are vv1, vv2.

Fig. 6 shows a detailed drawing of a sample architecture of the storage device B. The storage device B has a structure similar to that of the storage device A except that there is no volume virtualization function. Thus, the memory 53000 does not contain a virtual volume management program or a virtual volume management table. In this embodiment, the storage device B includes two data interfaces and three real volumes, but there can be any number of data interfaces and real volumes. Also, in the storage device B, the data interface IDs are d1, d2, and the volume IDs are vb1, vb2, and vb3.

Fig. 9 shows an example of the device detection list 13500 stored in the SAN management server 10000. Numbers arbitrarily assigned within the SAN management server are entered in the Detection ID field in Fig. 9. In the Device Type field, the type of the device in the SAN is entered. In the Device Information field, the vendor, the device name, and the like are entered. In the IP Address Information field, the address of each device in the management network 70000 is entered. The Volume Virtualization Function field indicates whether or not each device is equipped with the volume virtualization function. The Virtualization ID field is an entry in which storage is entered in the form of device information. If a real volume of the storage device is virtualized from another device, the detection ID of the other device performing the virtualization is entered. This information is set up ahead of time by a SAN administrator using the output module 15000 and the input module 16000 of the SAN management server 10000. The SAN manager 13100 uses this list to identify and communicate with the management agent of each device.

Fig. 10 shows a sample data interface management table stored in the server A 20000. The Data Interface ID field in Fig. 10 stores the ID of the data interface of the server. The Port WWN (World Wide Name) field stores the Port WWN, which serves as a unique identifier within the Fibre Channel, assigned to the data interface. The SCSI ID field stores an identifier (SCSI ID number) of the SCSI target device to which the data interface connects.

Fig. 11 shows a sample volume management table stored in the server A20000. The server A handles three volumes, and the server A uses the volume management table to store information about volumes that it is provided with. The LU (Logical

Unit) ID field in the volume management table stores an arbitrarily assigned volume that the server A handles.
The Data Interface ID field stored a data interface identifier in the server A used to access the volume. The SCSI ID field stores the SCSI ID number of the SCSI target device to which the data interface is connected. The LUN field contains the SCSI logical unit for accessing the volume in the SCSI target device. The Volume Information field contains the vendor name, device name, and the volume identifier of the device providing the volume to the server.

In the example in Fig. 11, the server A is provided with the real volume va1 from the storage device A and the virtual volume vv1, whose real volume is located in the storage B but which is virtualized by the storage device A. The server A is also provided with the real volume vb3 from the storage device B. The real volume vb3 of the storage device B is provided to the server A without being virtualized by the storage device A.

Fig. 12 shows a sample FC connection management table 33300 stored in the switch A30000. The FC connection management table stores information relating to the connection targets of s1 through s6, which are the data interfaces of the switch A30000. The Data Interface ID field of the FC connection management table stores the data interface ID of the switch A30000. The Switch Port WWN field stores the Port WWN of the data interface. The Target Port WWN field stores the Port WWN for the data interface of the server or storage device to which the data interface connects.

Fig. 13 shows examples of data interface management tables stored in storage devices. The data interface management table 43200 is a table in the storage device A and the data interface management table 53200 is a table in the storage device B. The Data Interface Field of the data interface management table stores the identifier of a data interface in the storage device. The Port WWN field stores the Port WWN of the data interface.

Fig. 14 shows examples of real volume management tables stored in storage devices. The real volume management table 23300 is in the storage device A, and the real volume management table 53300 is in the storage device B. The Real Volume ID field of the real volume management table stores the ID of a real volume in the storage device. The Path Availability field stores whether or not there is path for when another device accesses the real volume. The data interface ID field stores the identifier of the data interface in the storage device used to access the volume. The SCSI ID field stores the SCSI ID number assigned to the data interface. The SCSI LUN field stores the SCSI logical unit number used to access the real volume. If the Path Availability field for a real volume in the real volume management table indicates that there is no path, the real volume has not been used yet. Thus, the Data Interface ID field, the SCSI ID field, and the SCSI LUN field will all contain "N/A", indicating that these values have not been defined.

Fig. 15 shows an example of the virtual volume management table 43500 stored in the storage device A40000. First, the contents of the Virtual Volume fields will be described. The Virtual Volume ID field stores an arbitrarily assigned identifier for the virtual volume provided to the server. The Path Availability field stores whether or not a path is available for when another device accesses the virtual volume. The Data Interface field stores the identifier of the data interface in the storage device used to access the volume. The SCSI ID field stores the SCSI ID number assigned to the data

interface. The LUN field stores the SCSI logical unit used to access the real volume. In this embodiment, the storage device A 40000 provides the virtual volume vv1 by way of the data interface c1. The virtual volume vv2 is unused.

Next, the contents of the real volume fields in the virtual volume management table 43500. The Real Data Interface ID field stores the identifier of the data interface of the storage device A used to access the real volume serving as the virtual volume indicated by the identifier in the Virtual Volume ID field. The SCSI ID field stores the SCSI ID number of the SCSI target device to which the real data interface is connected. The LUN field stores the SCSI logical unit number used to access the volume provided by the storage device by way of the real data interface. The Real Volume Information field stores the name of the storage device providing the real volume accessed by way of the real data interface and the identifier and storage capacity of this real volume. This information can be obtained using the SCSI INQUIRY command and the like.

The virtual volume management table stores data only for the volumes that are virtualized by a storage virtualization device (e.g., storage device A). Thus, the real volumes vb1, vb2, which are in the storage device B and can be accessed by the data interface c2 are entered in the virtual volume management table since they are recognized by the storage device A, virtualized by the storage device A, and provided to a server as virtual volumes. Since the real volume va1, which can be accessed by way of the data interface c1, is provided to the server without being virtualized by the storage device A, information relating to the va1 is not entered in the virtual volume management table 43500. Information about the real volume vb3 in the storage device B is also not entered in the virtual volume management table 43500 since vb3 is directly recognized by the server A by way of the data interface d2. Thus, the virtualization devices supporting volume virtualization functions do not need to virtualize all real volumes in a storage device.

Fig. 16 shows examples of failure analysis dictionaries 13600 in the SAN management server 10000. Fig. 16 (a) is the failure analysis dictionary for the storage device A, Fig. 16 (b) is for the storage device B, Fig. 16 (c) is for the server A, Fig. 16 (d) is for the switch A. These dictionaries are used to analyze SNMP Trap messages issued from a device when a failure or the like takes place. The details of this operation will be described later. The Failure Code field stores the failure code in the Variable Bindings field of an SNMP Trap message. The Failure Component field stores the failure component associated with the failure code. The Identifier field stores an identifier indicating the failure component. The Reason field stores the reason the message was issued. The Severity field stores the Severity of the Trap in the Specific Trap Type field of the SNMP Trap message.

Fig. 17 shows a failure log 13700 in the SAN management server 10000. The failure log stores an event ID assigned when the SAN manager receives a failure notification message, a time when the failure took place, a device name for the source of the failure notification message, the failure code in the failure notification message, a real mapping ID for the mapping containing the component, a virtual mapping ID for the mapping containing the component, and relationships with other failure events.

Fig. 18 shows an example of a failure severity conversion table in the SAN management server 10000. In a failure notification operation, which includes a severity conversion function, performed by the SAN manager, described later, this conversion table is used to define common severities for failure messages from multiple devices

received by the SAN manager and to define the operations performed by the SAN manager in response to the common severities. This table is defined by a SAN administrator when setting up the SAN environment.
The failure severity conversion table stores a common severity for failure messages from multiple devices, the severities of each device corresponding to the common severity, and the operations performed by the SAN manager in response to the common severity. For example, in the case of Fig. 18, if the severity of the storage device A is "3" or if the severity is "4", "5", or "6" for the storage device B, the common severity is considered "3" in the SAN environment. In response, the SAN manager sends an SNMP Trap and an e-mail message to the SAN administrator containing failure message information relating to the storage device A only.
The severity conversion table is defined based on the configuration information of the SAN. For example, in the severity conversion table shown in Fig. 18, severity 3 for the storage A and severities 4 - 5 are associated with common severity 3, and common severity 3 is defined so that an SNMP Trap and an e-mail message to the SAN administrator are sent containing failure message information relating to the storage device A only. The reason for this is that the real volume in the storage B is virtualized by the storage A and provided to the server, with input/output requests sent between the storage B and the server going by way of the storage A. Thus, the definition links the severity of the storage A with the severity of the storage B and provides output only for the failure information of the storage A, which virtualizes the real volumes of storage A and storage B.

<Generation of virtual volume mapping and real volume mapping by the SAN manager>

Next, the generation of virtual volume mapping and real volume mapping performed by the SAN manager 13100 of the SAN manager server 10000 will be described. This operation is performed periodically by having the processor 11000 of the SAN manager server 10000 execute a program stored in the memory 13000. In this operation, a current virtual volume mapping and real volume mapping for the SAN environment are generated and output. This will be used by the failure investigation operation and notification operation described later. Unless explicitly indicated, the steps described below are performed by the SAN manager 13100.

Fig. 19 shows a flowchart 1700 presenting an overview of a real topology and virtual topology display operation executed by the SAN manager 13100. The SAN manager 13100 detects devices in the SAN based on the device detection list 13500, communicates with the management agent of each device, and copies the information stored in each device shown in Fig. 10 through Fig. 15 (step 1710). Next, the SAN manager 13100 stores the copied information in the real topology repository 13400 (step 1720). Then, using the information stored in step 1720, the virtual volume mapping table 13300 is generated (step 1730). Furthermore, the real volume mapping management table 13200, described later, is generated using the information in the real topology repository 13400 and the and the virtual volume mapping management table 13300 (step 1740). Finally, the results, such as the real topology, based on the contents of the virtual volume mapping management table 13300 and the real volume mapping management table 13200 are output (step 1750), and the operation is completed.

Fig. 20 is a flowchart showing the detailed operations performed by the SAN manager 13100 at the virtual volume mapping management table generation step 1730. Fig. 8 shows an example of the virtual volume mapping management table generated by the operation shown in Fig. 20.

For each of volume management table 23300 received from the servers and stored in the real topology repository 13400, the SAN manager 13100 performs the following operations on all the entries in the volume management table (step 1810).

First, the SAN manager generates a new entry in the virtual volume mapping management table and enters a newly allocated virtual mapping ID 13301. Then, the SAN manager enters a server name 13302 of the server that sent the volume management table 23300 being processed, a data interface ID 13304 stored in the volume management table, and a LU ID 13303 (step 1820). Next, the SAN manager checks the data interface of the switch to which the data interface 13304 is connected and enters a switch name 13305 and a data interface ID 13306.

More specifically, the SAN manager first uses the data interface 13304 of the server entered in the virtual volume mapping management table 13300 as a key to retrieve the Port WWN associated with the data interface ID in the data interface management table 23200 received from the server and stored in the real topology repository 13400. Then, the SAN manager uses this Port WWN as a key to look up the FC connection management table 33200 received from the switch A and stored in the real topology repository to determine which data interface of which switch the server is connected to. The result is entered as a target switch name and a target switch data interface ID 13306 (step 1830). As a result of this operation, information about the server is entered in the left half (server fields) of the virtual volume mapping management table 13300.

Next, the SAN manager performs operations to enter information in the right half of the virtual volume mapping management table 13300. The SAN manager uses the vendor name and the device name entered as volume information in the volume management table 23300 to determine if the volume entered in the volume management table is provided by a virtualization device. More specifically, the SAN manager determines if the device is equipped with the storage virtualization function by looking up the device detection list 13500 using the device name of the device name as a key. If the device is equipped with a virtualization function, it is assumed that the volume is provided from the virtualization device (step 1840). The operation branches in the following manner depending on the result from step 1840.

If the volume is provided by a storage device that is not a virtualization device, the SAN manager enters the device name and the volume ID from the volume information field in the volume management table 23300 as a storage name 13309 and a volume ID 13311 in the storage fields of the virtual volume mapping management table 13300. Then, the SAN manager obtains the ID of the data interface used to access the real volume by looking up the real volume management table received from the storage device using the entered volume ID 13311 as the key. The result is entered in the storage data interface ID 13310 (step 1860).

Then, the SAN manager checks the data interface of the switch to which the entered storage data interface 13310 is connected and enters the switch name and the data interface ID. More specifically, the SAN manager first determines the Port WWN of the storage data interface by looking up the data interface management table received from the storage device using the storage data interface ID 13310 as the key. Then, this

WWN is used as a key to look up the FC connection management table 33200 received from the switch A to determine which data interface of which switch the storage data interface is connected to. Then, the SAN manager enters the results as a target switch name 13307 and a target switch data interface ID 13308 (step 1870).

If step 1850 determines that the volume is provided by a virtualization device, the SAN manager performs the following operations. First, the SAN manager takes the device name and the volume ID entered in the volume information field of the volume management able 23300 and enters them as a storage name 13309 and a volume ID 13311 in the virtual volume mapping management table 13300. Then, the SAN manager determines the ID of the data interface of the storage device A used to access the volume by looking up the virtual volume management able 43500 and the real volume management table 43300 received from the virtualization device using the volume ID 13311 that was entered as the key. The result is entered as the storage data interface ID 13310 (step 1861). Next, the data interface of the switch associated with the data interface of the storage device A is checked, and the switch name and data interface ID are entered.

More specifically, the SAN manager checks the data interface ID of the switch associated with the data interface by looking up the data interface management table 33400 received from the switch A using the storage data interface ID 13310 as the key. The results are then enter as the target switch name 13307 and the target switch data interface ID 13308 (step 1871).

Step 1850 cannot determine exceptional device types if the volume management table 23300 is not stored in the real topology repository 13400 because the device is not entered in the device detection list 13500, or if the device is not equipped with a management interface. If, in this manner, information is not entered in the volume information fields of the volume management table 23300, the storage fields are left empty (step 1862).

The operation at step 1730 is completed when the above steps have been executed by the SAN manager for all entries of all volume management tables received by the SAN management server from the servers and stored in the real topology repository.

Fig. 21 shows the detailed flow of operations performed at the real volume mapping management table generation step 1740 by the SAN manager 13100. Fig. 7 shows an example of a real volume mapping management table generated by the operations shown in Fig. 21.

The SAN manager 13100 executes the following operations for each of the entries in the virtual volume mapping management table 13300 generated at step 1730 (step 1910).

First, the SAN manager creates a new entry and enters a newly allocated real mapping ID 13201 (step 1920). Next, the SAN manager uses the storage names 13309 of the entries in the virtual volume mapping management table 13300 to determine if the determine if the device indicated by the storage name is equipped with the virtualization function. More specifically, the SAN manager looks up the device detection list 13500 using the storage name 13309 as the key and checks if the device has the virtualization function.

If the device has the virtualization function, the SAN manager performs the following steps. The SAN manager looks up the virtual volume management table

43500 received from the device indicated by the storage name 13309 using the volume ID 13311 in the virtual volume mapping management table entry as the key to retrieve entries with a virtual volume ID matching the volume ID 13311 (step 1940). Next, for each entry obtained in this manner, the SAN manager prepares a real volume management mapping table (step 1950).

Then, the SAN manager copies the server field contents (fields 13302 through 13306) of the current entry in the virtual volume mapping management table 13300 to the server fields (fields 13202 through 13206) of the newly prepared entries. The real data interface ID in the real volume information field of the entry in the virtual volume management table 43500 retrieved in step 1940 is copied to the switch data interface ID field 13208 of the storage fields. The storage name and the volume name in the real volume information field for the entry in the virtual volume management table 43500 is copied to the storage name entry 13209 and the volume ID entry 13211 in the storage fields.

The content of the virtual mapping ID 13301 in the virtual volume mapping management table 13300 is copied to the associated virtual mapping ID field 13212. The content of the switch name 13307 in the storage fields of the virtual volume mapping management table is copied to the switch name field 13207 in the storage fields (step 1960). The SAN manager then retrieves the ID of the data interface to which this volume is connected by looking up the real volume management table received from the storage device using the volume ID entered in the volume ID 13211 in the storage fields of the real volume mapping management table 13200. This ID is entered in the storage data interface ID 13210 in the storage fields of the virtual volume mapping management table.

If step 1930 determined that there is no virtualization function, the SAN manager copies the current entry (fields 13302 through 13311) of the virtual volume mapping management table 13300 to the entry (fields 13202 through 13211) in the real volume mapping management table 13200. The entry in the virtual mapping ID 13301 in the virtual volume mapping management table 13300 is entered in the associated virtual mapping ID 13212 of the real mapping management table 13200.

The above operations fill entries 13201 through 13212 in the real volume mapping management table 13200.

When the above steps have been executed by the SAN manager for all entries in the virtual volume mapping management table 13300, the operation indicated in step 1740 is completed.

Fig. 23 shows an example of a real topology display output to the output module 15000 by the SAN manager 13100 based on the real volume mapping table shown in Fig. 7. A real topology display 2010 is a sample output based on the real volume mapping management table 13200 indicating the connections between servers, switches, and storage devices.

<Failure investigation operation performed by the SAN manager>

An example of the failure investigation operation performed by the SAN manager will be described.

Currently, failure monitoring functions performed by SAN management software often uses SNMP protocol Trap messages defined in RFC1157 ("A Simple Network Management Protocol (SNMP)") prepared by the IETF (Internet Engineering

Task Force). However, identifying the failure component down to the virtual volume level is difficult since the volumes allocated to the servers in virtual volume technology are virtualized. Also, having a SAN administrator investigate a failure requires that the SAN administrator has a high degree of knowledge about each device, thus making administration costs extremely high.

In this operation, the SAN manager receives failure notifications from multiple devices and analyzes the effect these failures will have on real volume and virtual volume I/O accesses based on SAN configuration information obtained from the management agents and stored in the real topology repository. Also, relations between failure messages are determined. The SAN manager analyzes the failure messages received over a fixed period preceding the receipt of the current failure message to determine if there are any relationships with the current failure message. The results from this are output by the SAN manager so that the burden on the SAN administrator from analyzing multiple failure messages and investigating failures is reduced.

Before presenting the failure investigation operation, the format as shown in Fig. 25 of an SNMP Trap message received from devices in the SAN by the SAN manager will be described, and an example of an SNMP Trap message will be described. An SNMP Trap message is a failure notification message sent to the SAN management server 10000 by a management agent in a device in the SAN. Fig. 25 (a) shows the format of an SNMP Trap message.

An SNMP message is formed from: a message header; a community name field for the message destination; a PDU (Protocol Data Unit) Type field indicating message type; an Enterprise field indicating the name of the sending device vendor; an Agent Address indicating the source IP address; a Generic Trap Type field indicating the type of the Trap message; a Specific Trap Type field indicating the specific code uniquely defined by the sending device vendor; a Timestamp field indicating the time at which the message was sent; and a Variable Bindings field storing the contents of the message uniquely defined by the sending device vendor. A PDU Type field value of "4" indicates that the message is an SNMP Trap message. A Generic Trap Type field value of "6" indicates that the Trap message is based on a vendor-specific definition of the sending device vendor. In such cases, the Trap message must be interpreted based on the individual vendor definition.

Fig. 25 (b) shows an example according to this embodiment of an SNMP Trap message sent by the storage A 40000 to provide notification of a hardware failure in its own hardware. The message shown in Fig. 25 (b) is recognized as an SNMP Trap message since the PDU Type is "4". The message is recognized as a Trap message based on vendor-specific definitions of the sending device vendor since the Generic Trap Type is "6". In this case, the Specific Trap Type field is defined by the vendor to contain the Severity of the failure and the Variable Bindings field to contain a failure code indicating the failure component. Thus, the SNMP Trap message in Fig. 25 (b) indicates that a failure has taken place with Severity of "1" and a failure code of "30c1".

Fig. 22 shows a flowchart 2400 illustrating an example of failure investigation operations performed by the SAN manager 13100 in the SAN management server 10000. Unless explicitly indicated otherwise, the steps described below are executed by the SAN manager 13100.

The SAN manager 13100 waits until an SNMP Trap message is received from a device (step 2410). When a message is received, the SAN manager extracts the IP

address of the device sending the message from the Agent Address field in the message (step 2415). The extracted IP address is used as a key to look up the device detection list 13500 stored in the real topology repository 13400 (step 2420).

If the IP address is not found in the device detection list 13500, the Trap message is from an unregistered device and therefore the SAN manager cannot analyze the contents of the Trap message. Thus, the SAN manager creates a new entry in the failure log 13700, assigns an event ID, and outputs the IP address as the failure device and the Trap message itself as the failure component (step 2465). Control then jumps to step 2455, described later.

If, at step 2420, the extracted IP address was in the device detection list 13500 and the device issuing the Trap message could be identified, the SAN manager checks to see if the SAN manager server 10000 is equipped with a failure analysis dictionary for the device (step 2425).

If a failure analysis dictionary is available at step 2425, the SAN manager creates a new entry in the failure log 13700, assigns an event ID, extracts the failure time from the Timestamp field in the message and enters it in the time field, and also enters the device name. Then, the failure analysis dictionary is looked up using the Variable Bindings field of the Trap message as the key. If the failure code is entered, the failure code is entered in the failure code field (step 2430).

If, at step 2425, a failure analysis dictionary is not available, the SAN manager creates a new entry in the failure log 13700, assigns an event ID, extracts the failure time from the Timestamp field in the message, enters it into the time field, and also enters the device name. Then, the SAN manager assumes that the failure component is the entire device, enters "entire device" in the failure code field, and continues on with the following steps (step 2431).

When step 2430 or step 2431 has been completed, the SAN manager determines if the failure component indicated by the failure code is associated with real volume mapping or virtual volume mapping (step 2435). More specifically, the failure code is used as the key to retrieve the failure component and its identifier from the entry in the failure analysis dictionary for the registered failure device name. Next, the failure device name and the failure component ID or the failure component obtained previously are used as keys to determine if there is a matching entry in the real volume mapping management table 13200. If there is a matching entry, the SAN manager extracts the real mapping ID 13201 and the virtual mapping ID 13212 from the entry and enters these into the real volume field and the virtual volume field in the entry being created in the failure log 13700.

Then, the SAN manager determines if the indicated failure component is associated with virtual volume management (step 2440). More specifically, the failure component ID or the failure component and the failure device retrieved at step 2435 are used as keys to see if there is a matching entry in the virtual volume management table 43500. If there is a matching entry, the SAN manager extracts a virtual volume ID from the entry. Then, the extracted virtual volume ID is used as the key to see if there is a matching entry in the real volume mapping management table 13200. The real mapping ID 13201 and the virtual mapping ID 13212 are extracted and entered in the real volume field and the virtual volume field in the entry being created in the failure log 13700.

After the relationship between the current entry in the failure log and real volume mapping and virtual volume mapping has been entered at step 2435 and step

2440, the SAN manager looks into relationships with other failure log entries. First, the SAN manager determines if the entry being created is for a hardware failure or an access error to another component (step 2445). More specifically, the failure code is used as the key to retrieve the reason for the failure in the failure analysis dictionary for the entered failure device name.

If the reason determined at step 2445 is a hardware failure, the event being created is assumed to be a "parent" event that may induce other failure events, and "parent event" is entered in the event relationship field (step 2450). If the reason found at step 2445 is an access error to another component, the event being created is assumed to be a "child event" that may have been issued due to another failure event, and "child event" is entered in the event relationship field (step 2451).

Finally, contents of the new failure log entry are output by the SAN manager as a failure message (step 2455). This concludes the description of the flowchart 2400.

A specific example of a failure investigation operation shown in the flowchart 2400 performed by the SAN manager will be described. Fig. 23 shows an example of how the failure log shown in Fig. 17 is output by the failure investigation operation in the flowchart 2400. Event IDs 1000, 1001, 1002, 1003 are four failure messages generated due to a hardware malfunction in the data interface ID d1 of the storage device B. How these four messages are analyzed and associated will be described.

When the failure message with the event ID 1000 is received, the SAN manager analyzes the event as a hardware malfunction in the data interface ID d1 of the storage device B at step 2430. Then, at step 2435, the SAN manager finds that there are relationships with the real volume mapping pm2 and the virtual volume mapping vm2. At step 2445, it is further determined that the hardware malfunction is a "parent event".

Next, when the failure message with the event ID 1001 is received, at step 2430 the SAN manager analyzes the event as an access error from when an I/O operation to the virtual volume vv1 in the storage device A was extended to the real volume vb1. Then, at step 2435, the SAN manager finds that there are relationships with the real volume mapping pm2 and the virtual volume mapping vm2. Then, at step 2445, the access error is determined to be a "child event". In a similar manner, the failure message with the event ID 1002 and the failure message with the event ID 1003 are determined to be "child events".

When outputting the failure message at step 2455, the SAN manager looks at the real volume fields, the virtual volume fields, and the event relationship fields of the failure messages issued over a fixed period to determine if there are any associations with these failure messages. If so, an identification is made as either "parent event" or "child event". The "fixed period" referred to here is a time interval specified by the SAN administrator and serves as a time unit used in associating failures. The event IDs 1000, 1001, 1002, 1003 in Fig. 17 are all associated with the real volume mapping pm2 and the virtual volume mapping vm2, and since it is also known that the event ID 1000 is a "parent event", these associations can be indicated in the failure event list window 2020 shown in Fig. 23, e.g., with a symbol 2021 in the event association field.

Also, as in event specification 2022, if the SAN administrator specifies a particular failure event, the SAN manager 10000 can graphically represent the real topology mapping associated with the specified event in the real topology display window 2010, e.g., as in the real topology mapping display 2011. Furthermore, the

contents of the specified event can be displayed in a manner that is easy to understand as in the failure notification window 2012.

Thus, by performing the failure investigation operation, the SAN manager can respond to failure messages from multiple devices in the SAN by analyzing the failure messages and automatically associating these messages with other failure messages, thereby reducing the burden on the SAN administrator of investigating failures.

<Failure notification operations performed by the SAN manager, including severity conversion>

An example of failure notification operations, including severity conversion, performed by the SAN manager will be described. In this operation, the SAN manager supports a severity conversion function for multiple storage devices connected to virtualization devices. Failure severity conversion table definitions defined by the SAN administrator ahead of time is used so that when a failure message is received, a high-level management program or an administrator is notified according to a common severity defined in the conversion table.

Fig. 24 shows a flowchart 2500 illustrating failure investigation operations performed by the SAN manager 13100 in the SAN management server 10000. Unless explicitly stated otherwise, the following steps are executed by the SAN manager 13100.

The SAN manager 13100 waits for an SNMP Trap message to be received from a device (step 2510). When a message is received, the SAN manager extracts the IP address of the device issuing the message from the Agent Address field in the message (step 2515).

Using the extracted IP address, the SAN manager determines if the device issuing the message is associated with the common severity definition (step 2520). More specifically, first the source of the message is identified by checking the device detection list 13500 to see if it contains the extracted IP address. Next, the failure severity conversion table 13800 is checked to see if there is a severity field associated with the identified device.

If, at step 2520, the device issuing the message is found to be not associated with the common severity definition, the SAN manager does not perform severity conversion and transfers the Trap message directly to the high-level management software (step 2526).

If, at step 2520, the device issuing the message is found to be associated with the common severity definition, the severity of the device issuing the message is extracted from the Specific Trap Type field in the SNMP Trap message (step 2525). The failure severity conversion table 13800 is looked up using the name of the device issuing the message and the extracted severity, and a common severity and an action are identified (step 2530). Finally, the action identified at step 2530 is executed (step 2535). This concludes the operations performed in the flowchart 2500.

A specific example of a failure notification operation performed by the SAN manager 10000 according to the flowchart 2500 will be described. In this example, an event with the event ID 2000 from the failure log shown in Fig. 17 is received. Step 2515 determines that the failure message with the event ID 2000 was sent by the storage device B, so at step 2520 it is determined that it is associated with the common severity definition. At step 2525, since the severity in the Trap message is "4", the action "Send storage A information as trap and e-mail" applies, and the failure message with the

event ID 2000 is not sent to the high-level management software or the SAN administrator.

Thus, the SAN manager performs the notification operation, including the severity conversion function, to provide a unified severity definition for failure messages received by the SAN manager from multiple storage devices, and the SAN manager can provide a failure notification function based on this definition.

The operation for creating the real topology mapping and virtual topology mapping for the storage network, the operation for investigating failures, and the operation for failure notification including severity conversion performed by the SAN manager all assumed that the storage device A 40000 is a virtualization device. However, the above operations can also be implemented in a configuration where a device other than the storage device A 40000 is a virtualization device that is connected to the management network 70000 and the SAN 60000.

With the embodiment described above, in a SAN equipped with a virtualization device, a device executing a SAN manager can receive failure messages from multiple devices in the SAN so that the SAN manager analyzes the failure messages and associates them with other failure messages in an automated manner, thus reducing the burden on the SAN administrator for investigating failures.

By defining unified severities for failure messages received by the SAN manager in a SAN from multiple storage devices and having the SAN manager perform failure notification based on these definitions, the SAN administrator and the high-level system management software can receive just the necessary failure information. This speeds up responses to failures after notification.

According to the present invention, when a failure message is issued from a device connected to the SAN, support for failure investigation can be provided to the SAN administrator.

Also, in the SAN, the SAN administrator and the high-level system management software can receive just the necessary failure information out of the failure messages issued from the devices in the SAN.